**УСТАНОВКА APACHE 2.4 + PHP 5.6 + MYSQL 5.6**

**НА WINDOWS 7/8/8.1**

Данная инструкция будет полезна начинающим WEB-программистам, которые собираются установить на своем ПК с Windows WEB-сервер Apache с интерпретатором PHP последних версий. Конечно же, намного проще установить сборку, наподобие Denwer и не париться по поводу настроек. Однако каждый WEB-программист просто обязан понимать процесс работы WEB-сервера, процесс его установки, настройки.

В статье описывается установка связки **Apache 2.4** + **PHP 5.6** + **MySQL 5.6** на операционную систему Windows 7 / 8 / 8.1

**Подготовительные работы**

Убедитесь, что у вас именно Windows 7 или новее. Если у Вас Windows XP, данная инструкция не поможет Вам, так как описанная здесь версия Apache не работает под Windows XP.

Удалите все WEB-серверы, которые устанавливали до этого, зайдите в службы и убедитесь, что там нет служб Apache или IIS. Если у вас уже установлен какой-либо веб-сервер, второй параллельно скорее всего не заработает вовсе.

Если у Вас есть Скайп, обязательно отключите в настройках использование 80 порта.

Для работы очень желательно иметь файловый менеджер, позволяющий создавать файлы с любыми расширениями, либо, что еще лучше, редактор кода, наподобие

[Sublime Text](http://www.sublimetext.com/) (  <http://www.sublimetext.com/> )

или

Notepad++.

Структура папок может быть различной, однако данная инструкция написана под определенную структуру и, если ей следовать, всё гарантированно заработает.  
Перед тем как начать, создайте на диске **D** папку  **server** , внутри которой создайте 5 подпапок: **apache**, **php**, **tmp**, **htdocs**, **logs**. Именно так. Без указания номеров версий.

В каталоге **htdocs** создаем файлы (при помощи файлового менеджера или редактора кода):

**index.html** с содержимым: **It works!**

**index.php** с содержимым: **<?php phpinfo();?>**

Если файлового менеджера / редактора кода нет, скачайте архив с этими двумя файлами [по этой ссылке](http://blog.denisbondar.com/wp-content/uploads/2014/12/index.zip) ( <http://blog.denisbondar.com/wp-content/uploads/2014/12/index.zip> ). Но все же, лучше обзавестись вышеописанными инструментами.

В системную переменную PATH добавляем:

|  |  |
| --- | --- |
|  | **;d:\server\apache;d:\server\apache\bin;d:\server\php;**  - v |

Если кто не знает, как найти то место, где меняется PATH,

прочитайте [вот этот комментарий](http://blog.denisbondar.com/post/apache24php56win7#comment-5722)

( [http://blog.denisbondar.com/post/apache24php56win7#comment-5722](http://blog.denisbondar.com/post/apache24php56win7" \l "comment-5722) ).

**Замечания!**

Приведенные в примерах конфигурационных файлов номера строк могут отличаться для других версий apache и php.  
Все команды, включая редактирование файлов, должны осуществляться от имени Администратора.

**Установка Apache 2.4**

1. Загружаем **Apache 2.4 binaries VC11** под свою систему по ссылке <http://www.apachelounge.com/download/VC11/>

( <http://www.apachelounge.com/download/VC11/> )

Скорее всего, у Вас 64-разрядная ОС, поэтому Вам понадобится

файл с именем **httpd-2.4.16-win64-VC11.zip**

1. Если нужны какие-то дополнительные модули, их можно загрузить там же (для базовой установки нет необходимости)
2. Разархивируем содержимое папки **Apache24** из скаченного архива в **d:\**server **\apache**. Обратите внимание, что в **d:\server\apache** нужно положить не папку Apache24 из архива, а ее содержимое. Инсталлировать ничего не требуется.
3. В файле **d:\**server**\apache\conf\httpd.conf**

изменяем значение **ServerRoot** на **“d:/**server**/apache”**  (*строка 37*)

и значение **DocumentRoot** (и **Direcroty**) на **“d:/**server**/htdocs”**  (*строки 242 и 243*).

Так же следует раскомментировать строку 218

и изменить ее на: **ServerName localhost:80**

1. Изменяем параметры хранения логов в том же файле (находим параметры и изменяем):

|  |  |
| --- | --- |
|  | ErrorLog "d:/server /logs/apache-error.log" (*строка 294*)  CustomLog "d:/server /logs/apache-access.log" common (*строка 323*) |
|  |  |

1. Устанавливаем сервис Apache (от имени Администратора). Открываем командную строку от имени Администратора и вставляем туда следующую строку:

**MSVCR110.dll**

- v

|  |  |
| --- | --- |
|  | **d:\server\apache\bin\httpd.exe -k install**  - v |

1. Следим за сообщениями об ошибках во время установки сервиса. Если все сделано верно, никаких ошибок быть не должно. Если у вас после выполнения строки не появилась снова командная строка, то вы что-то сделали неправильно. Просто воспользуйтесь функциями скопировать и вставить, чтобы не допускать ошибок при перепечатке.
2. Создаем на рабочем столе ярлык для **d:\server\apache\bin\ApacheMonitor.exe**

- V

и/или помещаем его в автозагрузку (для открытия окна автозагрузки в WIN8 необходимо нажать WIN+R, затем ввести **shell:Startup** и нажать ОК)

1. Запускаем ApacheMonitor. В системном трее появится ярлык. Нажимаем на него левой кнопкой, выбираем Apache24 -> Start.
2. В браузере заходим на <http://localhost/> – должны увидеть **It works!**
3. Если не увидели такой надписи, разбираемся, что пошло не так (читаем логи, гуглим, пытаемся самостоятельно разобраться с проблемой, раз уж решили разбираться в тонкостях работы веб-сервера)

**Установка PHP 5.6**

1. Загружаем последнюю версию **VC11 x86 Thread Safe** по ссылке <http://windows.php.net/download/>. Обратите внимание, что нужен именно  **VC11**  и именно  **Thread Safe** .

Файл, который Вам нужен, скорее всего, будет называться наподобие:

**php-5.6.11-Win32-VC11-x86.zip**

1. Извлекаем содержимое архива в  **d:\server\php** . Как в случае с Apache, инсталлировать ничего не требуется.
2. В файл   **d:\server\apache\conf\httpd.conf**  добавляем строки:

(🡪 строка 180)

|  |  |
| --- | --- |
|  | LoadModule php5\_module "d:/server/php/php5apache2\_4.dll"  AddHandler application/x-httpd-php .php  # Путь к файлу php.ini  PHPIniDir "d:/server/php" |

1. И изменяем значение  **DirectoryIndex**  на  **index.html index.php**  (*строка 276*)
2. Используя ApacheMonitor, перезапускаем Apache (Apache24 -> Restart)
3. Заходим браузером <http://localhost/index.php> и убеждаемся, что PHP работает.
4. Делаем копию файла d**:\server\php\php.ini-development** с именем d**:\server\php\php.ini**
5. Пользуясь поиском, находим, раскомментируем и изменяем параметры:

|  |  |
| --- | --- |
|  | extension\_dir = "d:/server/php/ext" (*string 736*)  sys\_temp\_dir = "d:/server/tmp" (*string 740*)  extension=php\_mysql.dll (*string 890*)  extension=php\_mysqli.dll (*string 891*)  extension=php\_openssl.dll (*string 893*)  date.timezone = Europe/Zaporozhye (*string 926*) |

1. Выполняем в командной строке **php -m** чтобы просмотреть список подключенных модулей.
2. Перезапускаем Apache, используя ApacheMonitor

**Виртуальные хосты в Apache**

1. Если на компьютере производится отладка нескольких проектов, то могут понадобиться виртуальные хосты (сайты). На примере будем рассматривать два виртуальных хоста: **s1.localhost** и **s2.localhost**. Для удобства будем создавать папки с такими же названиями, что совсем не обязательно в реальной системе.
2. Следует учесть, что при использовании виртуальных хостов, параметр **DocumentRoot** конфигурационного файла Apache более не имеет смысла. Основным хостом сервера (который будет доступен по адресу <http://localhost/>) теперь является **первый виртуальный хост** в конфигурационном файле виртуальных хостов!
3. Для начала необходимо добавить в файл c:\Windows\System32\drivers\etc\hosts строки:

|  |  |
| --- | --- |
|  | 127.0.0.1 s1.localhost  127.0.0.1 s2.localhost |

Делать это нужно от имени Администратора. Желательно после этого перезагрузиться, но на практике этого не требовалось. Если изменить файл не удается (права доступа), вы можете скопировать его на рабочий стол, изменить, а затем скопировать обратно.

1. Создаем папки для виртуальных хостов d**:\server\htdocs\s1.localhost**

и d**:\server\htdocs\s2.localhost**, в которых будут располагаться файлы. В каждой из папок создаем файл **index.html** с содержимым **S1** и **S2** соответственно (для контроля того, что все работает как нужно)

1. Затем создаем папки для логов: d**:\server\logs\s1.localhost** и **d:\server\logs\s2.localhost** — тут будут храниться логи каждого из сайтов. Вообще-то, логи можно хранить в одной папке, но я привык делать так — мне так более удобно.
2. Далее необходимо включить поддержку виртуальных хостов.

В файле **d:\server\apache\conf\httpd.conf** раскомментировать

строку **Include conf/extra/httpd-vhosts.conf (*строка 502*)**

1. Далее редактируем файл **d:\server\apache\conf\extra\httpd-vhosts.conf** – он должен выглядеть так (каждому виртуальному хосту свой блок VirtualHost):

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | <VirtualHost \*:80>      ServerAdmin webmaster@s1.localhost      DocumentRoot "D:/USR/www/s1.localhost"      ServerName s1.localhost      ServerAlias www.s1.localhost      ErrorLog "D:/USR/log/s1.localhost/error.log"      CustomLog "D:/USR/log/s1.localhost/access.log" common  </VirtualHost>    <VirtualHost \*:80>      ServerAdmin webmaster@s2.localhost      DocumentRoot "D:/USR/www/s2.localhost"      ServerName s2.localhost      ServerAlias www.s2.localhost      ErrorLog "D:/USR/log/s2.localhost/error.log"      CustomLog "D:/USR/log/s2.localhost/access.log" common  </VirtualHost> |
|  |  |

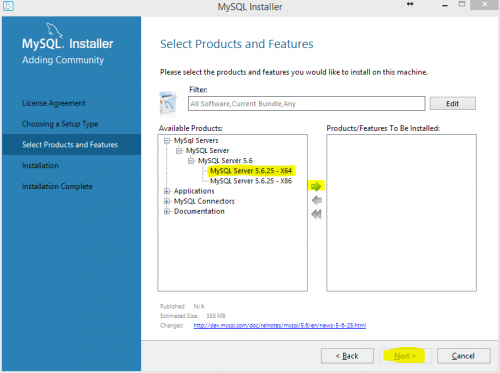
1. Первый виртуальный хост **s1.localhost** теперь будет основным хостом системы, т.к. его блок следует первым в конфигурационном файле, т.е. будет доступен при входе на <http://localhost/>
2. Перезапускаем Apache используя ApacheMonitor и заходим по очереди на <http://s1.localhost/> и <http://s2.localhost/>.
3. Итак. Для добавления нового виртуального хоста будет достаточно: создать папку для файлов хоста в D:\USR\www, создать папку для логов хоста в D:\USR\log, прописать имя хоста в файле hosts, добавить блок VirtualHost в файл конфигурации виртуальных хостов.

**Установка MySQL 5.6**

1. Загружаем MySQL Installer MSI (mysql-installer-community) по ссылке: <http://dev.mysql.com/downloads/windows/installer/5.6.html>

(необходима регистрация на сайте)

1. Во время установки на этапе **Choosing a Setup Type** выбираем: **Custom**, затем удаляем из устанавливаемых продуктов всё (в правой колонке), за исключением самого MySQL сервера (автоматически будут добавлены необходимые обязательные модули). Проще удалить все и добавить только сервер — все необходимые компоненты при этом добавятся сами

[](http://blog.denisbondar.com/wp-content/uploads/2014/11/mysql-win-choose.png)

1. Запускаем установку, нажав на **Execute** в следующем окне
2. После установки сервера нажимаем несколько раз **Next** и переходим к конфигурированию сервера
3. Выбираем тип **Development Machine**, опции оставляем как есть и нажимаем **Next**
4. Устанавливаем пароль основного пользователя **root** (обязательно) – здесь же можно добавить пользователей, но пока этого делать не нужно, нажимаем **Next**
5. Указываем, нужно ли сделать так, чтобы сервер запускался автоматически со стартом Windows (у меня автозапуск выключен, т.к. локальный MySQL-сервер используется только для отладки в очень редких случаях), нажимаем **Next**
6. Запускаем конфигурирование кнопкой **Execute**. Всё должно пройти успешно, после чего нажимаем **Next** и затем **Finish**.
7. Установка сервера MySQL завершена. Поздравляем!
8. Если вы оставили в настройках автозагрузку сервера, то более ничего делать не нужно. Сервер MySQL будет запускаться автоматически вместе со стартом операционной системы Windows. Иначе придется запускать сервер из окна **Службы** панели Управления компьютером Windows (или написать свой минискрипт для запуска и остановки службы). Я работаю на ноутбуке и постоянно запущенный сервер мне не нужен (расход ресурсов).
9. Для работы с MySQL сервером есть множество полезных и удобных утилит. Я рекомендую использовать [dbforge Studio for MySQL](http://www.devart.com/ru/dbforge/mysql/studio/). При помощи этого инструмента можно создавать пользователей БД и назначать права доступа. Возможно, есть более удобные и быстрые инструменты. Из всех, что я использовал, этот наиболее функциональный, хоть и не особо быстрый.